



LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

GSV Annotated Bibliography

R. S. Roberts, P. A. Pope, M. Jiang, T. Trucano, C. Aragon, K. Ni, T. Wei, L. Chilton, A. Bakel

September 14, 2010

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

GSV ANNOTATED BIBLIOGRAPHY

*Randy S. Roberts*¹, *Paul A. Pope*³, *Ming Jiang*¹, *Cecilia R. Aragon*⁴,
*Timothy G. Trucano*², *Kevin Ni*¹, *Thomas Wei*⁵, *Lawrence K. Chilton*⁶
and *Alan Bakel*⁵

{¹ Lawrence Livermore, ² Sandia, ³ Los Alamos, ⁴ Lawrence Berkeley,
⁵ Argonne, ⁶ Pacific Northwest} National Laboratory, USA

30 August 2010, v1.1

The following annotated bibliography was developed as part of the geospatial algorithm verification and validation (GSV) project for the Simulation, Algorithms and Modeling program of NA-22. Verification and Validation of geospatial image analysis algorithms covers a wide range of technologies. Papers in the bibliography are thus organized into the following five topic areas: Image processing and analysis, usability and validation of geospatial image analysis algorithms, image distance measures, scene modeling and image rendering, and transportation simulation models. Many other papers were studied during the course of the investigation including [1–17]. The annotations for these articles can be found in the paper “On the verification and validation of geospatial image analysis algorithms,” [18].

1 Image Processing and Analysis

The paper by Zitova and Flusser is an attempt at a comprehensive review of classic and recent image registration methods, regardless of the particular application [19]. A brief review of prior image registration survey papers is provided at the end of section 2 on page 980. Registration is defined as the geometric alignment of of a sensed image to a reference image. A universal registration method does not exist because of the wide variety of applications stemming from different viewpoints (multi-view analysis), different image acquisition times (multi-temporal analysis), different sensors (multi-modal analysis), and scene-to model registration, as well as combinations thereof. The majority of registration methods consist of four steps: feature detection (control object detection), feature matching (control object correspondence), transformation model estimation (mapping function parameter estimation), and image transformation/resampling. The paper consists mainly of these four topics, organized as sections with subsequent subsections: feature detection

(area-based methods, and feature-based methods); feature matching including area-based methods (correlation-like, Fourier, mutual information, optimization) and feature-based methods (spatial relations, invariant descriptors, relaxation, pyramids, wavelets); transformation model estimation (global, local, radial basis functions, and elastic methods); image transformation/resampling (forward vs. backward mapping, and nearest neighbor, bilinear, and cubic convolution). The paper concludes with a section on accuracy assessment and current trends/future outlook. (Note: This paper mentions an important point only in passing, specifically that the computer vision community is most advanced in interest point detection/identification methods while the remote sensing community is most advanced in geometric transformation methods; future developments should strive to combine the best from both communities).

The paper by Sutton, Stark and Bowyer reviews the concept of “generic” object models for general object recognition, i.e., recognition of an object category rather than a single object instance [20]. The predominant object description method in computer vision is to make use of a priori 3D information about the objects of interest in the form of geometric models. There are two types of “model-based” (or “CAD-based”) vision: object-centered and view-centered. Object-centered manipulates the 3D model’s pose until it aligns with the 2D image under study. View-centered summarizes an object’s feature visibility for all possible poses of the 3D model (e.g. an “aspect graph”). This second type of model-based vision is the most basic type of generic model, generally termed an “image feature configuration,” whereby all possible appearances of a 3D object in a 2D image are generalized. The next level of generic object model description sophistication is “3D shape generalization,” of which there are three types: parameterized geometric models, structural models, and parameterized structural models. The concept of geometric primitives assembled to create the whole object (a “part-whole” model) is included in this object description scheme. Finally, the highest level of generic object description is “function-based models,” in which an object category is defined in terms of which functional attributes an object must possess in order to belong to an object category. (Note: Although dated, this is a concise and clear description of object description in computer vision.)

The paper by Yao et al. describes a framework for creating a natural language description of an image [21]. Creating a textual description of an image is a very difficult problem. Although other papers have been published on automatically annotating imagery, i.e., automatically adding labels to salient objects in an image, the image-to-text problem has not received a great deal of attention in the literature. This is due, no doubt, to the difficulty of generating a natural language description from an image. The proposed system contains four major components: 1) An image parser that creates a parsed graph of the image, 2) a visual knowledge base that guides the image parser and serves as an image ontology to attach semantic meaning to the parse graph, 3) a general knowledge base that enhances the semantic representations of the parse graph, and 4) a text engine that translates the semantic representations into a natural language description.

Central to the author’s approach is a particular data structure called an And-Or graph (AoG), which is used as a representation of visual knowledge. It expresses an image grammar that specifies the composition, spatio-temporal and functional relationships between elements of the scene, and provides a visual vocabulary for the various parts of a scene. Using the AoG, content of parsed imagery is expressed in a semantic representation language such as the resource description language. In turn, the semantic representation can then be expressed in OWL, the web ontology language. At this point, the semantic content can be published on the semantic web, and allow semantic content tools to perform retrieval, processing and analysis of the image content.

2 Usability and the Validation of Geospatial Image Analysis Algorithms

In order to validate geospatial image analysis algorithms, we will need to measure their effectiveness in an operational environment. The ultimate success of the software will be dependent upon how effectively humans are able to use the tools to achieve the goal of detecting nuclear proliferation from geospatial imagery. Multiple studies of the search process in large image datasets have demonstrated that the most efficient procedures utilize a combination of algorithmic and human approaches – systems where sophisticated statistical learning and machine vision techniques are combined with a highly interactive visual interface to yield the most promising results in massive, ambiguous, and dynamic data. This methodology is known in the visualization field as *visual analytics* [22]. This is precisely the situation we are facing in evaluating geospatial image analysis algorithms. Additionally, a survey of current research in geospatial image analysis finds a cross-cutting theme of the importance of usability as a means of evaluating and determining the effectiveness of research algorithms. This points to the need to develop appropriate usability metrics and techniques for this process of evaluation, and to apply *usability engineering* techniques in our analysis. The papers and books listed below in an annotated bibliography of geospatial image analysis usability will assist us in the selection of such metrics and techniques.

We note that multiple publications in the geospatial image analysis field call for increased application of usability engineering and visual analytics. The term *geoanalytics* has been recently coined to describe portions of this field. *Usability engineering* is a term used to describe methods for analyzing and enhancing the usability of software, and was first coined by Jakob Nielsen in 1994 in a book of that title [23]. Usability engineering is the process of analyzing user needs, designing and prototyping interfaces that are grounded in users current practices, and producing effective results. This textbook describes applying systematic methods throughout the development lifecycle to increase interface usability. Usability is defined in the ISO 9241 standard as “the effectiveness, efficiency, and satisfaction with which specified users achieve specified goals in particular environments.” (ISO,

1997). Another definition outlined in the ISO 9126-1 standard (ISO, 2000) uses the term *quality in use*, which means the capability of the software product to enable specified users to achieve specified goals with effectiveness, productivity, safety and satisfaction within specified contexts of use.

Additionally, there are multiple papers discussing and evaluating visual analytics systems designed for use with geospatial imagery. To quote from Demsar in [24],

Classical data mining algorithms assume that data are independently generated and identically distributed. Geospatial data are multidimensional, spatially autocorrelated and heterogeneous. These properties make classical data mining algorithms inappropriate for geospatial data, as their basic assumptions cease to be valid. Extracting knowledge from geospatial data therefore requires special approaches. One way to do that is to use visual data mining, where the data is presented in visual form for a human to perform the pattern recognition. When visual mining is applied to geospatial data, it is part of the discipline called exploratory geovisualization.

Both automatic and visual data mining have their respective advantages. Computers can treat large amounts of data much faster than humans, while humans are able to recognize objects and visually explore data much more effectively than computers. A combination of visual and automatic data mining draws together human cognitive skills and computer efficiency and permits faster and more efficient knowledge discovery. Demsar presents an analysis of several systems where the task is finding selected images in geospatial data. Usability analysis is a primary method in the evaluation of these systems, and the paper describes in detail the methods used.

The book by Thomas and Cook defines and motivates visual analytics and lays out a research agenda for software design, development, and evaluation [22]. Visual analytics is the science of analytical reasoning facilitated by interactive visual interfaces. Visual analytics tools and techniques are used to synthesize information and derive insight from massive, dynamic, and ambiguous data. This includes visual representations and interaction techniques that take advantage of the human eye's broad bandwidth pathway into the mind to allow users to see, explore and understand large amounts of information at once.

The next few papers also focus on the theme of usability as a requirement for evaluation of geospatial image analysis software, and describe gaps in research and practice. The highly cited paper by Slocum et al. argues that the framework for evaluating the effectiveness of geovisualization methods should be based on cognitive theory and usability guidelines [25]. To quote from the paper,

Developments in hardware and software have led to (and will continue to stimulate) novel methods for visualizing geospatial data. It is our belief that these novel methods will be of little use if they are not developed within a theoretical

cognitive framework and iteratively tested using usability engineering principles. We argue that cognitive and usability issues should be considered [when] evaluating the effectiveness of geovisualization methods.

Slocum et al. propose the application of usability engineering extending beyond the traditional practice of user testing. Software effectiveness should be evaluated throughout its lifecycle, including design, development, and deployment. The authors point out the difficulty of defining the nature of users and their tasks, and call for an interdisciplinary effort involving geographic information scientists, cognitive scientists, usability engineers, computer scientists, and others.

Christophe and Inglada identify the profound gap between cutting edge algorithms to analyze satellite imagery described in the literature and methods available in production software, and calls for an approach to increase software usability [26]. The authors note that constraints on production impede the uptake of new research algorithms. This paper proposes an open source architecture to facilitate the transfer of new algorithms to production, called the Orfeo Toolbox. The framework provides for robust transfer and scalability of research algorithms to production. Initial feedback has been positive and the authors suggest that this framework can increase the usability of cutting edge algorithms.

The paper by Albu et al. is a survey of vision-based Human-Computer Interaction (HCI) systems for: Image-guided diagnosis, therapy planning, surgery, motor-impaired patient assistance, and support of the elderly [27]. This paper makes a strong case that end-user requirements have (or should have) a significant impact on the algorithmic design of the computer vision techniques (e.g. segmentation) underpinning HCI systems. Sections and subsequent subsections are: vision-based interfaces for enhanced data visualization (HCI for computer-aided diagnosis and therapy planning in clinical environments, research-oriented graphical interfaces, towards HCI design for collaborative and remote image analysis, and HCI for enhanced visualization during image-guided surgery); vision-based interfaces for the operating room (markerless tracking, and haptic devices for training); perceptual interfaces for motor-impaired users (gesture recognition), vision-based intelligent systems for elderly assistance (intelligent computer vision-based sensing agent).

3 Image Distance Measures

A fundamental operation in validating geospatial image analysis algorithms is comparing the output of the algorithm to a referent or benchmark. Given the wide variety of algorithms that might be brought to bear on geospatial data, the output of the algorithm can range from another image to a set of feature vectors. Comparing algorithm output to referent is essentially measuring the distance between the two quantities. However, it is not apparent as to what kinds of distance measures are appropriate for algorithm verification.

Distance measures, with respect to imagery, are typically found in either the study of image distortion or in image search applications. Thus, it becomes necessary to study these approaches, and apply applicable techniques to GSV problems. Our annotated bibliography of this area begins with three papers. The first paper describes a similarity measure based on fuzzy logic and distance measures that incorporate results from psychophysics experiments with the HVS. Since the paper was written, it has been cited in the technical literature approximately 200 times. The second paper describes a methodology of matching distance measures to ensembles of benchmark imagery using the distribution of differences between benchmark images. The last paper describes an approach to measuring the quality of a distorted image, given a reference image, using the image structure. This paper has been cited approximately 650 times since publication.

The focus of the paper by Santini and Jain is to develop a similarity measure based on fuzzy logic that can be used for content-based image retrieval applications [28]. The proposed similarity measure exhibits features that match experimental findings of how humans perceive similarity in imagery. The paper begins by observing that image similarity rests on two elements: finding a suitable set of features that appropriately encodes the characteristics that are measured, and a metric in the feature space to describe the distance between features. The authors note that a Euclidean metric space is often assumed, and that this assumption is often an uncritical decision. A discussion of similarity theories is next provided. The authors review the metric axioms for Euclidean spaces, point out the questionable properties of these axioms with respect to human perception, and then proceed to review a set of replacement axioms. These replacement axioms provide a more general framework for similarity measures, and form the basis for the feature contrast model, a set-theoretical similarity measure. Unfortunately, the feature contrast model is difficult to apply to imagery, and the authors proceed to present modifications that alleviate these issues.

The authors next launch into the development of a fuzzy set-theoretic measure that mitigates difficulties in applying the feature contrast model to imagery. This section includes several definitions and proofs as well as employing Choquet integrals to model interactions between various quantities. Upon completion of the theory section, they present results for two cases: a stick-figure face and texture patterns. The authors admit that the results are not conclusive, but the examples illustrate the comparable performance of the fuzzy set-theoretical similarity measure compared to several competing measures.

The paper by Wang et al. proposes a structural similarity metric that can be used to predict perceived image quality. The paper focuses on full-reference image quality assessment, that is, an assessment where a complete reference image is available for comparison [29]. Two full-reference quality metrics in common use are based on error sensitivity, and they are the mean-squared-error (MSE) and the peak signal-to-noise ratio (PSNR). While they are simple to calculate and use, they are not well matched to perceived image quality. Two images can have the same MSE (and therefore PSNR), but have very different types

of errors, some of which can be more objectionable than others. Quality assessments based on error sensitivity typically split the reference and test image into channels, compute the errors within each channel, and pool the errors often using the Minkowski norm. There are several problems with this approach, and most are rooted in the philosophy of equating image errors with loss of visual quality. The authors also note that natural images are highly structured, and these dependencies contain important information about the structure of the image. The Minkowski error metric is based on pointwise signal differences, and is independent of the underlying image structure.

The approach proposed by the authors finds a more direct way to compare the structure of the reference and test images. They espouse a philosophy that considers image degradation as a perceived change in structural information. Moreover, they assert that their approach more closely mimics the HVS. The authors illustrate their philosophy by devising a measure of structural similarity between images. The authors define structural information in the image as those attributes that represent objects in the image independent of the average illumination. Their approach segregates the task of computing the similarity measure into computing three functions: luminance comparison, contrast comparison and structure comparison. These three components are combined to yield the Structured SIMilarity (SSIM) Index. (The structure comparison function is essentially the correlation coefficient between the reference and test images.) The SSIM index is applied to each pixel in the image pair (reference and test) via a sliding window. The resulting image provides a spatially varying map of quality degradation. An overall measure of image quality is found by averaging the SSIM values over the image quality degradation map. Several experimental results are presented using JPEG and JPEG2000 images with various degrees of compression. The authors observe that many image quality assessment algorithms are consistent when applied to distorted images derived from the same reference image, but the quality assessment degrades when applied to sets of imagery created from different reference images or have a variety of different distortions. The SIMM quality metric was applied to several sets of imagery compressed using JPEG and JPEG2000, and was found to yield better consistency with perceived image quality when compared to PSNR and several other measures.

4 Scene Modeling and Image Rendering

Synthesizing and modeling 3D scenes is an important aspect of generating benchmark imagery data sets for verification and validation of geospatial data processing algorithms. Rendering imagery from these synthetic 3D scenes is an important technology for the Simulation, Algorithm and Modeling portfolio, because it enables the creation of precise imagery data sets that can enable high fidelity evaluation of geospatial algorithms. In these next two sections, we discuss scene modeling and image rendering.

4.1 Scene Modeling

In computer graphics, 3D modeling is the process of developing a mathematical representation of 3D objects and scenes, using geometric primitives and texture images. 3D models can be created automatically or manually, and there are numerous modeling techniques, including constructive solid geometry, implicit surfaces, and subdivision surfaces, just to name a few. For the test and evaluation of geospatial algorithms, we are interested in modeling large-scale terrains and cityscapes. There exist a wealth of computer graphics papers related to synthesizing terrains and modeling cities. We have selected four seminal papers from among them: “The Synthesis and Rendering of Eroded Fractal Terrains” [30], “Synthetic Topiary” [31], “Procedural Modeling of Cities” [32] and “Instant Architecture” [33].

Synthesizing terrain data is an important application area in computer graphics. Musgrave et al. [30], from Yale University, present a novel approach to synthesizing fractal terrain height fields using a two-pass approach: terrain generation and erosion simulation. The first pass generates a fractal terrain surface of varying smoothness and asymmetry, and the second pass incorporates a physical erosion model which simulates the hydraulic and thermal erosion processes. Unlike previous approaches based on fractional Brownian motion, their new approach, termed noise synthesis, provides arbitrary local control of fractal dimension and other statistical characteristics, such as crossover scale and lacunarity. An important contribution of their work is the use of erosion models to simulate the effects of running water (hydraulic erosion) and falling rocks (thermal weathering). While hydraulic erosion creates valleys and drainage networks, thermal weathering wears down steep slopes and creates talus slopes at their feet.

The paper first describes the problems with traditional fractal terrain models based on fractional Brownian motion: (1) the statistical characteristics of the surface are the same everywhere and (2) the surface lacks global erosion features due to isotropy and stationarity. Next, the paper describes the terrain synthesis process via noise synthesis using the Perlin noise function and how to modulate crossover scale and exercise local control over lacunarity. Then, the paper describes the physical erosion models: hydraulic erosion, which involves depositing water on vertices of the height field and allowing the water and sediment to move to lower vertices, and thermal weathering, which is a relaxation process that adjusts the altitude at each vertex based on its neighboring values and a user specified talus angle. Finally, the paper describes a fast ray tracing technique for height fields in general, termed *grid tracing*, and concludes with examples of rendered imagery produced from the proposed techniques.

Plant development is generally difficult to model in a realistic way, especially when environmental factors are taken into account. Existing models can be considered as either structure-oriented, which assumes that the developmental process is under endogenous control from within (e.g., adjacent branches on the plant), or space-oriented, which emphasizes

exogenous controls from the surrounding environment (e.g., roots growing around obstacles). Prusinkiewicz et al. propose a new approach, based on an environmentally-sensitive extension to the L-systems, which combines exogenous and interactive endogenous controls [31]. (A Lindenmayer system, or L-system, is a formal grammar commonly used for modeling plant growth.) An environmentally-sensitive L-system incorporates various attributes, such as position and orientation, into the derivation steps. During interpretation, query modules are used to provide values for these attributes. The authors were able to apply the system to simulate plant response to pruning and, in the process, create models of sculptured plants.

The paper begins by explaining the dichotomy between structured-oriented and space-oriented models for plant development. Then, the paper presents the formalism for L-systems and introduces the notion of environmentally-sensitive L-systems that utilize query modules in the derivation steps to access various attributes. To help explain the formalism, the paper presents a series of examples with increasing complexity and relevance to plant growth. Next, the paper describes a stochastic tree model based on this system that can more realistically approximate branching rate and crown area. Using this model, the paper describes the simulation of tree response to pruning and a flexible way to control the bifurcation frequency of reiterated branches. Finally, the paper presents a gallery of synthetic topiary, which is the art of clipping suitable plants into ornamental shapes, to highlight the efficacy of the proposed system.

Modeling and visualizing large-scale cityscapes is still a great challenge for computer graphics. The difficulty lies in being able to model the diversity of street patterns, buildings, forms and textures of an urban area in a (semi-) automatated manner. Parish and Mueller, from ETH Zuerich, present a novel system, called CityEngine, using a procedural approach based on L-systems to model cities [32]. The CityEngine system is a pipeline of consisting of the following steps: generate a network of highways and streets, divide the land into lots, and create geometry for buildings on the allotments. CityEngine is capable of modeling a complete city using only a small set of statistical and geographical input data, such as population density and land-water boundaries. By extending the L-system to utilize global goals and local constraints, the addition of new rules, such as different transportation networks and alternative land uses, can be very easily incorporated into the system.

The paper first describes what makes modeling a city difficult and why existing approaches, such as using aerial imagery to extract the buildings and streets via computer vision methods, are insufficient. Next, the paper describes the proposed system, CityEngine, which consists of the following steps: roadmap creation via the extended L-system, division of lots via subdivision, building generation via the L-system, and geometry and texture creation. Then, the paper describes how to create the street map using the extended L-system, which modifies the parameters to the production rules of the L-system based on global goals (e.g., road patterns) and local constraints (e.g., road intersections). Next, the paper describes subdividing the street map into allotments for buildings, and generating the

geometry (stochastic L-system) and the textures (semi-procedurally) of buildings. Finally, the paper concludes with imagery generated from CityEngine of a virtual Manhattan.

For urban reconstruction, the ability to model architectures with non-trivial design styles in an automated manner is a hard problem. Although design grammars, such as L-systems, have been used to model plants and cityscapes, they cannot be easily adapted to model buildings since they simulate growth in space, whereas buildings have stricter spatial constraints. Wonka et al. in [33] propose a framework based on shape grammars that can model a variety of architectural designs utilizing a large database of grammar rules, without the need to create individual grammars for each object to be modeled. Because a large database of grammar rules can produce instances that are incongruous, they introduce a new type of design grammar, called “split grammar,” to restrict the type of rules allowed. They also introduce a parameter matching system that enables users to specify high-level design goals and a control grammar to handle spatial distribution of design ideas corresponding to various architectural styles. In order to use the system effectively, however, a user must become familiar with the concept of grammars.

Wonka et al. first describes the difficulty of using existing design grammars, such as L-systems, for modeling buildings with non-trivial architectural designs. Next, the paper proposes a new framework for automatic building modeling that consists of the split grammar to derive shapes, the attribute-matching system to assign attributes, and the control grammar to enforce design patterns. The paper then provides formal definitions for shapes, grammars and the split grammar, which is a specialized type of set grammar operating on shapes with deterministic derivation. Next, the paper introduces the control grammar for propagating attributes spatially based on architectural principles and the attribute-matching system for selecting rules using a deterministic matching function to create candidate rules and a stochastic selection function to select the rule. Finally, the paper concludes by demonstrating a variety of buildings modeled using the proposed framework.

4.2 Image Rendering

Image rendering is the term used for computer generation of an image from a 3D model. Rendering techniques have been under development since the late 1960’s when computer graphics began in earnest. Roughly speaking, rendering technologies can be divided into two categories: photorealistic rendering and non-photorealistic rendering. Photorealistic rendering strives to produce imagery that appears to be real-life, while non-photorealistic rendering produces imagery that might be found in art, technical drawings and cartoons. Although imagery required to test and evaluate geospatial algorithms need not be photorealistic, the rendering technologies associated with photorealism are appropriate starting points. The Wikipedia website [34] has a listing of important papers published on image rendering covering the time span of 1968 to 2002. A second source is a list of seminal graphics papers compiled in 1998 [35]. From these lists, we have selected four papers of

significance “Distributed ray tracing” [36], “The rendering equation” [37], “Measuring and Modeling Anisotropic Reflection” [38] and “Global Illumination using Photon Maps” [39].

Ray tracing is one of the fundamental techniques in computer graphics for generating physical effects, such as shadows, reflections, and refracted light. However, the imagery produced from conventional ray tracing looks unrealistic due to their sharp quality. Cook, et.al, from Lucasfilm, points out that “[r]ay traced images are sharp because ray directions are determined precisely by geometry,” and by distributing spatially oversampled rays, fuzzy edges are easily rendered [36]. They introduced the idea of applying the Monte Carlo method to ray tracing, in order to “soften” the sharp physical effects produced by the conventional technique. Rather than using a single ray to sample many different domains, they average multiple rays distributed over an interval. They are able to achieve much more photorealistic effects such as motion blur, depth of field, penumbras, translucency, and fuzzy reflections. The paper concludes with several examples of their technique. The disadvantage with this approach is that the computational cost increases with the number of rays.

The paper first describes the shading function and how distributed ray tracing can overcome the problems with traditional simplifying assumptions and generate photorealistic effects such as glossy reflection, translucency, and penumbras. Next, the paper explains the depth of field phenomenon, the problem with the existing post-processing approach for producing this effect, and how to compute it accurately using distributed ray tracing. In a similar fashion, the paper then describes the motion blur phenomenon in animation, the problems with existing approaches for producing this effect, and how to overcome these problems by distributing the sample points in time. Finally, the paper summarizes all the physical phenomena that can be modeled using this approach, and concludes with examples of rendered images illustrating these effects.

Kajiya was the first to generalize a variety of known rendering algorithms into a single integral equation [37]. The equation is based on the law of conservation of energy: the equilibrium radiance leaving a point is given as the sum of emitted plus reflected radiance under a geometric optics approximation. In other words, at each position and direction, the outgoing light is the sum of the emitted light and the reflected light, and the reflected light is the sum of the incoming light from all directions. This equation provides the foundation for various realistic rendering techniques in computer graphics, and the formalism for evaluating the tradeoffs between computational cost and the achievable photorealism of the rendered effect.

The paper begins by presenting the rendering equation and describing the terms in the equation. Kajiya calls the terms *transport quantities* and derives the relationships between the transport quantities and standard radiometric quantities. He goes on to present several approximate solutions to the integral equation which in turn are shown to be various rendering algorithms such as a classical method for rendering shading surfaces, the classical ray tracing approximation, distributed ray tracing [36] and the radiosity approximation.

The author next presents a Markov chain approach to solve the rendering equation. He finally develops several variance reduction techniques that are applied to the rendering equation. The paper concludes with several examples of rendered imagery produced by application of the rendering equation and solution techniques developed in the paper.

The interaction of light with a surface can be expressed as a bidirectional reflectance distribution function (BRDF), which is a 4D function that computes the ratio of reflected radiance to the incident irradiance at an idealized surface point. BRDFs are based on either empirical or theoretical models: empirical models can produce BRDFs with nonphysical tuning parameters and theoretical models are only approximations to real reflectance. Ward developed a BRDF model that is both physically valid and fits the measured reflectance data for both isotropic and anisotropic distributions [38]. With a lack of practical devices that can capture a comprehensive BRDF measurement of an anisotropic surface, they built an imaging gonioreflectometer using a half-silvered plastic hemisphere and a CCD camera with a fisheye lens. He used a Gaussian distribution to model isotropic reflectance and an elliptical Gaussian to model anisotropic reflectance, which were capable of describing most significant reflection phenomena.

Ward first outlines the problems plaguing existing empirical and theoretical models for BRDFs and points out the reason for the popularity of the Phong model, which is neither theoretically plausible nor empirically correct, is its mathematical simplicity. After formally defining BRDF, the paper then describes how to measure BRDF of a surface using a conventional gonioreflectometer, and the overall design and calibration of the proposed imaging gonioreflectometer that can be used to measure anisotropic reflectance. Next, the paper presents a new mathematical model that can fit the measured data: elliptical Gaussian distributions. In order to render (unbiased with low variance) the anisotropic reflection using the proposed model, the paper uses a hybrid deterministic and stochastic ray tracing algorithm. Finally, the paper concludes by showing BRDFs measured and fitted for various materials and the renderings produced from them.

Simulating global illumination is essential for synthesizing photorealistic images in general environments. Jensen introduced photon mapping as a two-pass global illumination algorithm for solving the rendering equation [39]. The first pass traces photons through the scene and records their interactions with surfaces in a photon map. Two photon maps are produced from this pass: a high resolution caustics photon map to render caustics that are visualized directly and a low resolution global photon map as a rough approximation of the lighting within the scene. The second pass renders the final image using distributed ray tracing in which the radiance for each pixel is estimated using k-nearest neighbor samples from the photon maps. Effects such as subsurface scattering of light in translucent materials and particulate matter such as smoke and water vapor have been successfully modeled using photon maps. For computational speedup, a GPU implementation of photon mapping has been introduced.

Jensen begins by describing the shortcomings of two popular global illumination algo-

gorithms: (1) Monte Carlo ray tracing, which is time consuming and produces noisy results and (2) radiosity, which is memory consuming and cannot properly handle specular reflection. Next, the paper presents the two-pass algorithm. The first pass constructs the photon maps by emitting a large number of photons from the light sources in the scene, and the second pass renders the final image by solving the rendering equation using photon maps in distributed ray tracing. The paper describes how various components of the render equation, such as direction illumination, specular reflection, caustics, and soft indirect illumination can be addressed using the photon maps. Then the paper describes how to estimate the radiance leaving a surface in a given direction by using the photon map and the surface BRDF. Finally, the paper compares results from photon mapping and Radiance software, which uses path tracing.

5 Transportation Simulation Models

Traffic modeling and simulation is important to the development of a validation platform. Inference engines can be validated against scenarios with realistic simulated traffic in cases where real vehicle movement data is unavailable. TRANSIMS and SUMO are two ongoing software development projects for traffic simulation [40, 41]. Both are modular systems incorporating multimodal transportation (travel of individuals by cars, busses, trains, etc). TRANSIMS is focused on simulating vehicle movement by way of defining population demographics and characteristics. SUMO, however, is only focused on the actual simulation of vehicle movement. Demand for individual vehicle movement or collective vehicle flows is a direct input in SUMO, whereas in TRANSIMS individual vehicle movement demand is generated from the demographic data. Both simulators use an iterative method where routes are planned, then simulated in the transportation network, recalculated to account for traffic congestion, and re-simulated. There has been major development on both these tools since the introductory papers that have added features and capabilities. However the basic principles of the two simulators have not changed.

The TRansportation Analysis and SIMulation System (TRANSIMS) was developed for the Department of Transportation and Environmental Protection Agency. Smith et al. summarizes the original development of TRANSIMS [40]. The original objective of TRANSIMS was to develop realistic models and simulations of a regional transportation network for the purposes of evaluating pollution, traffic congestion, and studying many other problems. It utilizes models of population demographics to generate traffic patterns consistent with real life population. The simulation is done on the individual traveler level as opposed to the zonal level.

TRANSIMS originally consisted of four major modules that produce a realistic simulation of traffic. The first module is the *Household and Commercial Activity Disaggregation* module which uses census data as well as data from other sources to create a baseline

synthetic population of households. Households are characterized by a probabilistic combination of demographic information, e.g. income, age, and number of workers in a family. This demographic information is used to generate activity demand and travel behavior which feeds into the second module, the *Intermodal Route Planner*. This second module develops a trip plan for each individual. The trip plan has a level of acceptability based on an individual’s socio-economic status and purpose, e.g. a rich person would want the shortest and safest path to work. The Planner also accounts for various preferences that each individual preferences such as departure time or congestion avoidance.

The planner feeds into the *Transportation Microsimulation Module* which takes the trip plans for each traveler and attempts to execute their trip plan on the transportation system. The transportation system is multimodal and uses a complex road network that includes signalized road networks and different types of streets. The result of this microsimulation can feed back into the Intermodal Route Planner so that the Planner can recalculate the trips that do not meet an individual’s acceptability requirements. The consideration of a continuous representation of vehicle position and the cellular discrete representation is discussed in [40]. Current TRANSIMS capability has settled on the usage of a cellular discrete representation of vehicle position. Finally the *Environmental Modeling* module translates overall traveler behavior into measures of environmental quality, e.g. emissions or air quality.

Krajzewicz et al. describes the initial work for the open-source traffic simulator SUMO (Simulation of Urban MObility) in [41]. The focus of the SUMO project is to provide an open source, and thereby easily extensible, platform for continuous, microscopic and multi-modal traffic simulation. SUMO is designed to be fast and exact on multiple platforms. The purpose of providing an open source system is to allow for researchers to examine the underlying model of the simulation. This also allows for the extension of the simulations through new modules and models as needed by researchers. SUMO is continuous in both space and time unlike TRANSIMS; it is microscopic in the sense that traffic flows are modeled based upon individually placed vehicles. And, it is multi-modal in the sense that the simulation can model public transport system and alternative train networks. To allow for multi-modal transportation, the atomic unit is a single human described by a desired departure time and destination. In this way, the route of a human can be made up of multiple subroutes of one mode.

SUMO also incorporates tools to allow for the generation of a highly complex road network, as the developers do not expect the user to define the road network manually [41]. For example, the network importer/converter can assign turn lanes and other intersection properties based upon the priorities of the edges entering a junction. The network import tool has since added several types of allowable input formats including OpenStreetMaps. Like TRANSIMS, SUMO’s method of route planning is an iterative process where routing and traffic simulation are repeated several times to account for changes in traffic congestion and speeds with each new routing plan.

6 Acknowledgements

The authors would like to acknowledge the support of Dr. Alexander Slepoy, Program Manager, Simulations, Algorithms, and Modeling; Office of Nonproliferation Research & Development, National Nuclear Security Administration.

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. LLNL-XXX. v11

References

- [1] Software Engineering Standards Committee of the IEEE Computer Society, *IEEE Guide for Software Verification and Validation Plans*, ser. IEEE Std 1059-1993. Institute of Electrical and Electronics Engineers, 1994.
- [2] Navy Modeling and Simulation Management Office, *Modeling and Simulation Verification, Validation, and Accreditation Implementation Handbook*, ser. Volume 1 VV&A Framework. United States Department of the Navy, 30 Mar 2004.
- [3] American Institute of Aeronautics and Astronautics Staff, *AIAA Guide for the Verification and Validation of Computational Fluid Dynamics Simulations*. American Institute of Aeronautics & Astronautics, 1998.
- [4] American Society of Mechanical Engineers, *V&V 10 - 2006 Guide for Verification and Validation in Computational Solid Mechanics*. American Society of Mechanical Engineers, 2006.
- [5] United States Department of Energy, “ASCI Software Quality Engineering: Goals, Principles and Guidelines,” United States Department of Energy, Tech. Rep. DOE/DP/ASC - SQE-2000-FDRFT-VERS2, Feb 2001.
- [6] R. Klein *et al.*, “ASC predictive science academic alliance program verification and validation whitepaper,” Lawrence Livermore National Laboratory, Tech. Rep. UCRL-TR-220711, 2006.
- [7] D. Desovski *et al.*, “Verification and validation of a fingerprint image registration software,” *EURASIP Journal on Applied Signal Processing*, vol. 2006, Article ID 15940, 9 pages, 2006.
- [8] L. Fei-Fei, R. Fergus, and P. Perona, “Learning generative visual models from few training examples: An incremental Bayesian approach tested on 101 object categories,” *Computer Vision and Image Understanding*, vol. 106, no. 1, pp. 59–70, 2004.

- [9] F. Tanner, B. Colder, C. Pullen, D. Heagy, M. Eppolito, V. Carlan, C. Oertel, and P. Sallee, "Overhead imagery research data set - an annotated data library & tools to aid in the development of computer vision algorithms," in *Proceedings of the 2009 IEEE Applied Imagery Pattern Recognition Workshop (AIPRW)*, Oct 14-16 2009, pp. 1-8.
- [10] N. Pinto, D. Cox, and J. DiCarlo, "Why is real-world visual object recognition hard?" *PLoS Computational Biology*, vol. 4, no. 1, pp. 0151-0156, Jan 2008.
- [11] J. Ponce, T. L. Berg, M. Everingham, D. A. Forsyth, M. Hebert, S. Lazebnik, M. Marszalek, C. Schmid, B. C. Russell, A. Torralba, C. K. I. Williams, J. Zhang, and A. Zisserman, "Dataset issues in object recognition," in *Toward Category-Level Object Recognition*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2006, vol. 4170/2006.
- [12] G. Hay and T. Blaschke, "Special issue: Geographic object-based image analysis (GEO-BIA)," *Photogrammetric Engineering and Remote Sensing*, vol. 76, no. 2, pp. 121-122, 2009.
- [13] L. Obrst, W. Ceusters, I. Mani, S. Ray, and B. Smith, "The evaluation of ontologies: Toward improved semantic interoperability," in *Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences*, C. Baker and K. Cheng, Eds. Springer, New York, NY, 2007.
- [14] M. Kavouras and M. Kokla, *Theories of Geographic Concepts: Ontological Approaches to Semantic Integration*. CRC Press, Boca Raton, Florida, 2008.
- [15] W. Ceusters, B. Smith, and J. Flanagan, "Ontology and medical terminology: Why description logics are not enough," in *Towards an Electronic Patient Record (TEPR 2003)*, May 10-14 2003, unpaginated CD.
- [16] T. Gruber, "Ontology," in *Encyclopedia of Database System*, L. Liu and T. M. Özsu, Eds. Springer-Verlag, 2008. [Online]. Available: <http://tomgruber.org/writing/ontology-definition-2007.htm>
- [17] X. Jiang, C. Marti, C. Irniger, and H. Bunke, "Distance measures for image segmentation evaluation," *EURASIP Journal on Applied Signal Processing*, vol. 2006, Article ID 35909, 10 pages, 2006.
- [18] R. Roberts, T. Trucano, P. Pope, C. Aragon, M. Jiang, T. Wei, L. Chilton, and A. Bakel, "On the verification and validation of geospatial image analysis algorithms," in *Proceedings of the 2010 IEEE International Symposium on Geoscience and Remote Sensing*, Honolulu HI, July 26-30, 2010.

- [19] B. Zitova and J. Flusser, “Image registration methods: a survey,” *Image and Vision Computing*, vol. 21, pp. 977–1000, 2003.
- [20] M. Sutton, L. Stark, and K. Bowyer, “A short survey: What is a ‘generic’ object model for computer vision?” in *Proceedings of the 5th Florida Artificial Intelligence Research Symposium*, 1992, pp. 252–256.
- [21] B. Yao, X. Yang, L. Lin, M. Lee, and S. Zhu, “I2T: Image parsing to text description,” *IEEE Proceedings*, vol. 98, no. 8, pp. 1485–1508, August, 2010.
- [22] J. J. Thomas and K. A. Cook, *Illuminating the Path: The Research and Development Agenda for Visual Analytics*. National Visualization and Analytics Ctr, (accessed 30-August-2010). [Online]. Available: <http://nvac.pnl.gov/agenda.stm>
- [23] J. Nielsen, *Usability Engineering*. San Francisco: Morgan Kaufmann, 1994.
- [24] U. Demšar, J. Krisp, and O. Křemenová, “Exploring geographical data with spatio-visual data mining,” *Progress in Spatial Data Handling: 12th International Symposium on Spatial Data Handling*, pp. 149–166, 2006.
- [25] T. A. Slocum, C. Blok, B. Jiang, A. Koussoulakou, R. Koussoulakou, D. R. Montello, S. Fuhrmann, and N. R. Hedley, “Cognitive and usability issues in geovisualization,” *Cartography and Geographic Information Science*, vol. 28, no. 1, pp. 61–75, 2001.
- [26] E. Christophe and J. Inglada, “Open source remote sensing: Increasing the usability of cutting-edge algorithms,” *IEEE Geoscience and Remote Sensing Society Newsletter*, pp. 9–15, 2009.
- [27] A. Albu, “Vision-based user interfaces for health applications: A survey,” in *Lecture Notes in Computer Science*, G. Bebis *et al.*, Eds. Springer-Verlag, 2006.
- [28] S. Santini and R. Jain, “Similarity measures,” *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 21, no. 9, pp. 871–883, sep 1999.
- [29] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *Image Processing, IEEE Transactions on*, vol. 13, no. 4, pp. 600–612, april 2004.
- [30] F. Musgrave, C. Kolb, and R. Mace, “The synthesis and rendering of eroded fractal terrains,” in *Proceedings of the 16th Annual Conference on Computer Graphics and Interactive Techniques*. ACM, 1989, pp. 41–50.
- [31] P. Prusinkiewicz, M. James, and R. Měch, “Synthetic topiary,” in *ACM SIGGRAPH Conference*, 1994.

- [32] Y. Parish and P. Müller, “Procedural modeling of cities,” in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. ACM, 2001, pp. 301–308.
- [33] P. Wonka, M. Wimmer, S. François, and W. Ribarsky, “Instant architecture,” *ACM Transactions on Graphics*, vol. 22, no. 3, pp. 669–677, 2003.
- [34] Wikipedia, “Rendering (computer graphics),” (accessed 30-August-2010). [Online]. Available: [http://en.wikipedia.org/wiki/Rendering_\(computer_graphics\)](http://en.wikipedia.org/wiki/Rendering_(computer_graphics))
- [35] R. Wolfe, ““Seminole” graphics,” *SIGGRAPH Comput. Graph.*, vol. 32, no. 3, pp. 46–47, 1998.
- [36] R. L. Cook, T. Porter, and L. Carpenter, “Distributed ray tracing,” *SIGGRAPH Comput. Graph.*, vol. 18, no. 3, pp. 137–145, 1984.
- [37] J. T. Kajiya, “The rendering equation,” *SIGGRAPH Comput. Graph.*, vol. 20, no. 4, pp. 143–150, 1986.
- [38] G. J. Ward, “Measuring and modeling anisotropic reflection,” in *ACM SIGGRAPH Conference*, 1992, pp. 265–272.
- [39] H. W. Jensen, “Global illumination using photon maps,” in *Seventh Eurographics Workshop on Rendering*, 1996, pp. 21–30.
- [40] L. Smith, R. Beckman, and K. Baggerly, “TRANSIMS: Transportation analysis and simulation system,” Los Alamos National Lab., NM (United States), Tech. Rep., 1995.
- [41] D. Krajzewicz, G. Hertkorn, C. Rössel, and P. Wagner, “SUMO (Simulation of Urban MObility); An open-source traffic simulation,” in *4th Middle East Symposium on Simulation and Modelling (MESM2002)*. sn, 2002, pp. 183–187.